3 Методология тестирования на проникновение

Одним из важнейших факторов, влияющих на успешность проведения теста на проникновение, является стандартная методология испытания. Отсутствие стандартных методик проведения теста на проникновение означает отсутствие однотипности. Мы уверены, вы не хотите быть испытателем, проводящим бессистемный тест, применяя то один, то другой инструмент и не имея представления о том, какие результаты этот тест должен принести.

Методология — это набор стандартных правил, практических действий и процедур, которые реализуются при работе с любой программой, предназначенной для проверки информационной безопасности. В методологии тестирования на проникновение в первую очередь определяется план проведения теста. В этом плане предусматриваются не только цели проведения испытаний, но и действия, которые должны быть выполнены для оценки истинного состояния безопасности сети, приложений, системы или любой их комбинации.

Испытатель обязан обладать практическими навыками проведения испытаний. Он должен владеть инструментами, с помощью которых проводится тест. Только четко определенная методика проведения испытаний на проникновение, теоретические знания и практические навыки испытателя позволят провести полный и достоверный тест на проникновение. Но в то же время методология не должна препятствовать испытателю анализировать свои догадки.

Технические условия

В этой главе для работы вам понадобятся установленная ранее операционная система Kali Linux и приложение Nmap.

Методология тестирования на проникновение

Чтобы определить, какой тест вам сейчас нужно будет провести, необходимо знать, какие тесты существуют, в каких областях и для каких целей они применяются. Все тесты можно разделить на три группы.

- **Методы «белого ящика».** В этой группе тестов испытатель хорошо знает проверяемую систему и имеет полный доступ ко всем ее компонентам. Испытатели работают с клиентом и имеют доступ к закрытой информации, серверам, запу-

щенному программному обеспечению, сетевым схемам, а иногда даже к учетным данным. Этот тип испытаний обычно проводится для проверки новых приложений перед их вводом в эксплуатацию, а также для регулярной проверки системы в рамках ее жизненного цикла — *Systems Development Life Cycle (SDLC)*. Такие мероприятия позволяют выявить и устранить уязвимости раньше, чем они могут попасть в систему и навредить ей.

- **Методы «черного ящика».** Эта группа тестов применима, когда испытателю ничего не известно об испытуемой системе. Этот тип тестирования в наибольшей степени похож на настоящие атаки злоумышленника. Испытатель должен получить всю информацию, творчески применяя имеющиеся у него в распоряжении методы и инструменты, но не выходя за рамки заключенного с клиентом соглашения. Но и этот метод имеет свои недостатки: хотя он и имитирует реальную атаку на систему или приложения, испытатель, используя только его, может пропустить некоторые уязвимости. Это очень дорогой тест, так как занимает большое количество времени. Выполняя его, испытатель изучит все возможные направления атаки и только после этого сообщит о результатах. Кроме того, чтобы не повредить проверяемую систему и не вызвать сбой, испытатель должен быть очень осторожным.
- **Методы «серого ящика».** Тест учитывает все преимущества и недостатки первых двух тестов. В этом случае испытателю доступна только ограниченная информация, позволяющая провести внешнюю атаку на систему. Испытания обычно выполняются в ограниченном объеме, когда испытатель немного знает о системе.

Для обеспечения наилучших результатов тестирования, независимо от применяемых тестов на проникновение, испытатель должен соблюдать методологию проведения испытаний. Далее мы более подробно обсудим некоторые наиболее популярные стандартные методы проведения испытаний.

- Руководство по тестированию OWASP.
- Руководство по тестированию на проникновение PCI.
- Стандарт выполнения тестирования на проникновение.
- NIST 800-115.
- Руководство по методологии тестирования безопасности с открытым исходным кодом (OSSTMM).

Руководство по тестированию OWASP

Open Web Application Security Project (OWASP) — этот проект объединил разработчиков программных средств с открытым исходным кодом. Люди, входящие в данное сообщество, создают программы для защиты веб-приложений и веб-сервисов. Все приложения создаются с учетом опыта борьбы с программами, наносящими вред веб-сервисам и веб-приложениям. OWASP — это отправная точка для системных архитекторов, разработчиков, поставщиков, потребителей и специалистов по

безопасности, то есть всех специалистов, которые принимают участие в проектировании, разработке, развертывании и проверке на безопасность всех веб-сервисов и веб-приложений. Другими словами, OWASP стремится помочь создавать более безопасные веб-приложения и веб-сервисы. Главным преимуществом руководства по тестированию OWASP является то, что по представленным результатам тестов можно получить всестороннее описание всех угроз. Руководство по тестированию OWASP определяет все опасности, которые могут повлиять на работу как системы, так и приложений, и оценивает вероятность их появления. С помощью описанных в OWASP угроз можно определить общую оценку выявленных проведенным тестированием рисков и выработать соответствующие рекомендации по устранению недостатков.

Руководство по тестированию OWASP в первую очередь сосредотачивает внимание на следующих вопросах.

- Методы и инструменты тестирования веб-приложений.
- Сбор информации.
- Проверка подлинности.
- Тестирование бизнес-логики.
- Данные испытаний.
- Тестирование атак типа «отказ в обслуживании».
- Проверка управления сессиями.
- Тестирование веб-сервисов.
- Тест AJAX.
- Определение степени рисков.
- Вероятность угроз.

PCI-руководство по тестированию на проникновение

Здесь собраны нормативы для компаний, соответствующих требованиям PCI (Payment Card Industry — индустрия платежных карт). Причем в руководстве вы найдете нормативы не только по стандарту PCI v3.2. Оно создано Советом безопасности по стандартам PCI, в котором определены методы тестирования на проникновение в рамках программ управления уязвимостями.

Стандарт *PCI Data Security Standard (PCI DSS)* версии 3.2 был выпущен в апреле 2016 года *Советом по стандартам безопасности индустрии платежных карт* (PCI SSC). После обновления стандарта требования были уточнены, появились дополнительные указания и семь новых требований.

Для устранения проблем, связанных с нарушениями секретности личных данных владельцев карт, а также для защиты от существующих эксплойтов в стандарт PCI DSS V. 3.2 были включены различные изменения, большинство из которых относятся к поставщикам услуг. В эти изменения были добавлены новые требования к тестированию на проникновение, согласно которым тестирование с сегментацией для поставщиков услуг выполнялось по крайней мере каждые шесть месяцев или

после любых значительных изменений в элементах управления/методах сегментации. Кроме того, в этом стандарте содержится несколько требований, обязывающих поставщиков услуг в течение года непрерывно отслеживать и поддерживать критически важные элементы управления безопасностью.

Стандартное проведение тестов на проникновение

Стандарт выполнения тестирования на проникновение состоит из семи основных разделов. Они охватывают все требования, условия и методы проведения испытаний на проникновение: от разведки и до попыток проведения пентестов; этапы сбора информации и моделирования угроз, когда, чтобы добиться лучших результатов проверки, испытатели работают инкогнито; этапы исследования уязвимостей, эксплуатации и пост-эксплуатации, когда практические знания испытателей в области безопасности соединяются с данными, полученными в ходе проведения тестов на проникновение; и как заключительный этап — отчетность, в которой вся информация предоставляется в виде, понятном клиенту.

Сегодня действует первая версия, в которой все стандартные элементы испытаны в реальных условиях и утверждены. Вторая версия находится в стадии разработки. В ней все требования будут детализированы, уточнены и усовершенствованы. Поскольку план каждого теста на проникновение разрабатывается индивидуально, в нем могут быть применены разные тесты: от тестирования веб-приложений до проведения испытаний, предусмотренных для тестирования методом «черного ящика». С помощью этого плана сразу можно определить ожидаемый уровень сложности конкретного исследования и применить его в необходимых, по мнению организации, объемах и областях. Предварительные результаты исследования можно увидеть в разделе, отвечающем за сбор разведданных.

Ниже в качестве основы для выполнения тестов на проникновение приведены основные разделы рассматриваемого нами стандарта.

- Предварительное соглашение на взаимодействие.
- Сбор разведданных.
- Моделирование угроз.
- Анализ уязвимостей.
- Эксплуатация.
- Пост-эксплуатация.
- Составление отчета.

NIST 800-115

Специальное издание *Национального института стандартов и технологий* (National Institute of Standards and Technology Special Publication, NIST SP 800-115) является техническим руководством по тестированию и оценке информационной безопасности. Публикация подготовлена *Лабораторией информационных технологий* (Information Technology Laboratory, ITL) в NIST.

В руководстве оценка безопасности трактуется как процесс определения того, насколько эффективно оцениваемая организация отвечает конкретным требованиям безопасности. При просмотре руководства вы увидите, что в нем содержится большое количество информации для тестирования. Хотя документ редко обновляется, он не устарел и может послужить в качестве справочника для построения методологии тестирования.

В этом справочнике предлагаются практические рекомендации по разработке, внедрению и ведению технической информации, тестам безопасности и процессам и процедурам экспертизы, охватывая ключевой элемент или техническое тестирование на безопасность и экспертизу. Данные рекомендации можно использовать для нескольких практических задач. Например, поиск уязвимостей в системе или сети и проверка соответствия политике или другим требованиям.

Стандарт NIST 800-115 предоставляет большой план для испытаний на проникновение. Он позволяет убедиться, что программа тестирования на проникновение соответствует рекомендациям.

Руководство по методологии тестирования безопасности с открытым исходным кодом

OSSTMM — документ, довольно сложный для чтения и восприятия. Но он содержит большое количество актуальной и очень подробной информации по безопасности. Это также самое известное руководство по безопасности на планете с примерно полумиллионом загрузок ежемесячно. Причина такой популярности в следующем: эти инструкции примерно на десятилетие опережают все остальные документы в индустрии безопасности. Цель OSSTMM — в развитии стандартов проверки безопасности Интернета. Данный документ предназначен для формирования наиболее подробного основного плана для тестирования, что, в свою очередь, обеспечит досконального и всестороннее испытание на проникновение. Независимо от других организационных особенностей, таких как корпоративный профиль поставщика услуг по тестированию на проникновение, это испытание позволит клиенту убедиться в уровне технической оценки.

Фреймворк: общее тестирование на проникновение

Несмотря на то что стандарты различаются по количеству условий, тестирование на проникновение можно разбить на следующие этапы.

1. Разведка.
2. Сканирование и перечисление.
3. Получение доступа.
4. Повышение привилегий.
5. Поддержание доступа.

6. Заметание следов.
7. Составление отчета.

Рассмотрим каждый этап более подробно.

Разведка

Это первый и очень важный этап в тесте на проникновение. На него может уйти немало времени. Многие испытатели делят данный этап на две части: активную и пассивную разведку. Я же предпочитаю эти два этапа объединить, так как полученные результаты скажут сами за себя.

Разведка (рекогносцировка) — это систематический подход, когда вы стараетесь обнаружить расположение и собрать максимально возможное количество информации о целевой системе или машине. Это еще называется *сбором следов*.

Для проведения данного процесса могут быть использованы следующие методы (в действительности список методов может быть значительно шире).

- Социальная инженерия (это увлекательный метод).
- Исследование в Интернете (с помощью поисковых машин Google, Bing, LinkedIn и т. д.).
- Путешествие по мусорным бакам (можно испачкать руки).
- Холодные звонки.

Вы можете выбрать любой из перечисленных методов для получения информации о целевой системе или машине. Но что же мы все-таки должны на данном этапе узнать?

Нам, конечно, может быть полезным каждый бит информации. Но у нас должна быть приоритетная цель. При этом учтите, что собранные данные, которые на текущем этапе могут показаться ненужными, позже могут пригодиться.

Сначала для нас будет очень важна следующая информация.

- Имена контактов в организации.
- Где располагается организация (если такие данные есть).
- Адреса электронной почты (эти данные можно использовать позже для фишинга, то есть сбора конфиденциальных данных).
- Номера телефонов важных персон, работающих в этой компании (пригодятся для фишинга).
- Операционные системы, используемые в компании, например Windows или Linux.
- Объявления о работе.
- Резюме сотрудников (прошлое и настоящее).

На первый взгляд все эти данные кажутся полезными (разве что смущают объявления о работе). Но представим, что вы встречаетесь с системным администратором. Зная основные требования, вы можете получить большое количество

информации о внутренней системе организации. Это можно использовать для разработки направления атаки.

Для этих же целей служат и резюме сотрудников. Зная, что люди умеют делать, легко можно определить, с какими системами они работают, а какие им недоступны.

Вам это может показаться утомительным. Но имейте в виду: чем больше информации вы соберете, тем больше у вас будет возможностей для принятия решений как сейчас, так и позже.

Мы считаем, что к разведке следует прибегать на протяжении всего взаимодействия.

Сканирование и перечисление

Без сомнения, почти каждый специалист по безопасности хочет сразу заняться эксплуатацией. Но без понимания основ, эксплойтов и, самое главное, среды, в которой они находятся, этот шаг не принесет никакой пользы и даже может спровоцировать ошибки или, что еще хуже, разрушение среды.

Сканирование и перечисление позволяют испытателю на проникновение понять среду целевой системы. Результат, полученный в ходе этих проверок, предоставит *красной команде* отправную точку для использования уязвимостей в разных системах.

Термин red team (красная команда) взят из военной среды и определяет «дружественную» атакующую команду. В противовес ей существует команда защитников — blue team (голубая команда). При работе красной команды снимаются все ограничения и производится реальная атака на инфраструктуру: от атак на внешний периметр до попыток физического доступа, «жестких» социотехнических тестов (тест с использованием методов социальной инженерии).

Сканирование — это поиск всех доступных сетевых служб (TCP и UDP), работающих на целевых узлах. Оно может помочь красной команде обнаружить, может ли быть на целевой машине открыт SSH/Telnet. В этом случае, используя систему грубой силы, можно попытаться войти через него. Тогда мы можем обнаружить файловые ресурсы для загрузки данных с уязвимых сайтов или принтеров, на которых могут храниться имена пользователей и пароли. *Перечисление* — это обнаружение служб в сети, что позволит нам лучше понять информацию, полученную от сетевых служб.

Сканирование

Если вы не знаете, включен ли брандмауэр, задействована ли система обнаружения вторжений и производится ли мониторинг целостности файлов, идеально подходит полный тест на проникновение. При сканировании можно обнаружить отдельные уязвимости. В этом случае при тестировании на проникновение будет предпринята

попытка проверить, можно ли обнаруженные уязвимости использовать в целевой среде. Рассмотрим все типы сканирования.

ARP-сканирование

С помощью широковещательного запроса мы можем получить преимущество в добыче информации об IP-адресе. Каждый широковещательный кадр ARP запрашивает, у кого какой IP-адрес. При этом запрашиваемый IP-адрес при каждом запросе увеличивается на единицу. После того как хост получит этот IP-адрес, он даст ответ, сопоставив запрошенный IP-адресом соответствующий ему MAC-адрес. ARP-сканирование является быстрым и эффективным методом и обычно не вызывает никаких аварийных сигналов. Только есть проблема: ARP — протокол второго уровня и поэтому не может перейти границы сети. То есть, если красная команда находится в сети, например, по адресу 192.100.0.0/24, а ваша цель (цели) — в сети 10.16.X.0/24, вы не сможете отправлять ARP-запросы для 10.16.X.0/24.

Сетевой картограф (Nmap)

Nmap является главной ищейкой в сканировании портов и перечислении. Мы не сможем в данной книге описать все параметры и модули Nmap. Вместо этого мы рассмотрим сканы, которые чаще всего используют при тестировании.

Но сначала расскажем, в каком состоянии может быть порт.

- *Открыт*. Приложение на целевом компьютере прослушивает соединения/пакеты на этом порту.
- *Закрыт*. Порт в данное время не прослушивает ни одно из приложений, но может быть открыт в любое время.
- *Фильтр*. Брандмауэр, фильтр или другое сетевое препятствие блокирует порт таким образом, что Nmap не может определить, открыт он или закрыт.

В Nmap нам доступны следующие параметры:

- `O` — обнаружение ОС;
- `p` — сканирование порта;
- `p-` — сканирование всех портов (от 1 до 65 535);
- `p 80,443` — сканирование портов 80 и 443;
- `p 22-1024` — сканирование портов от 22 до 1024;
- `top-ports X` — здесь в качестве `X` указывается число наиболее используемых портов, которые мы будем сканировать. Чтобы ускорить сканирование, мы обычно указываем значение `100`;
- `sV` — обнаружение служб;
- `Tx` — определение скорости сканирования;
- `T1` — очень медленное сканирование портов;
- `T5` — очень быстрое сканирование портов (с большим шумом);

- `sS` — скрытное сканирование;
- `sU` — сканирование UDP;
- `A` — определения версии ОС, сканирование с использованием сценариев и трассировка.

Сканирование портов/TCP-сканирование в Nmap. Эта служба запускается путем активации соединения (`SYN`) на каждом порте целевого хоста. Если порт открыт, хост ответит (`SYN, ACK`). Соединение закрывается (`RST`), если команда отправлена инициатором (рис. 3.1).

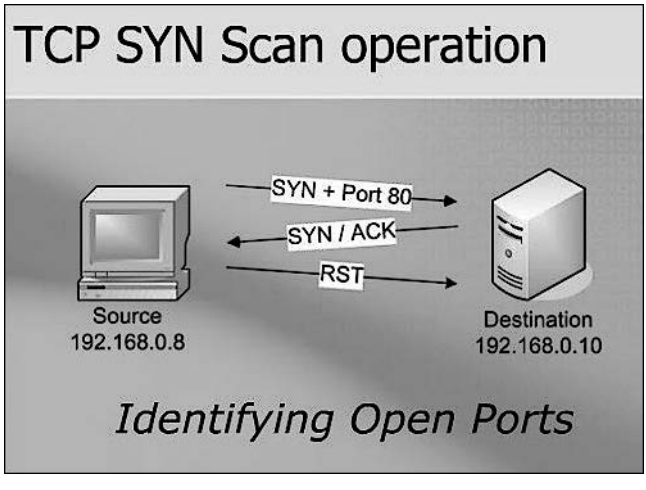

Рис. 3.1. Операция сканирования TCP SYN

Полуоткрытое/скрытое сканирование в Nmap. Этот параметр запускается путем отправки соединения (`SYN`) на каждый порт целевого хоста. Если порт открыт, хост на запрос ответит (`SYN, ACK`). Если порт закрыт, хост ответит сбросом соединения (`RST`). Если ответ не получен, можно предположить, что порт фильтруется. Разница между TCP- и скрытым сканированием заключается в том, что инициатор соединения не возвращает пакет подтверждения (`ACK`). Эффективность такого сканирования в том, что регистрируется только полностью установленное соединение.

Обнаружение OS в Nmap. Данный параметр использует различные методы для определения типа и версии операционной системы. Это очень полезно для обнаружения уязвимостей. Поиск версии ОС покажет в операционной системе известные уязвимости и эксплойты. Для этого введите следующую команду:

```
nmap 172.16.54.144 -O
```

Обнаружение служб в Nmap. Как и при обнаружении ОС, этот параметр пытается определить службу и версию, как показано на рис. 3.2:

```
nmap 172.16.54.144 –sV
```

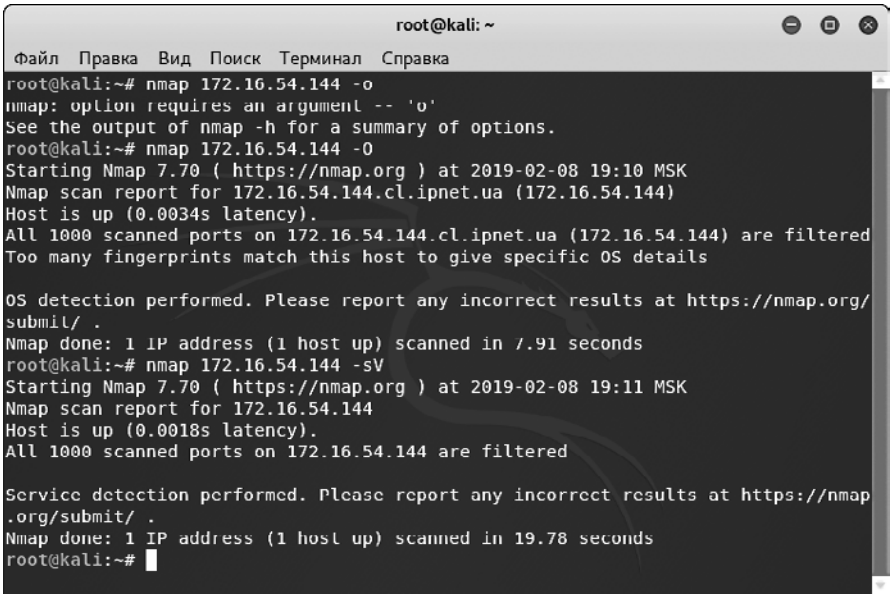

Рис. 3.2. Обнаружение служб

Nmap ping sweeps (Пинг-разведка Nmag). Этот параметр обрабатывает каждый IP-адрес в заданном диапазоне. Если узел подключен и настроен для ответа на запросы ping, он выдаст ICMP-ответ (рис. 3.3).

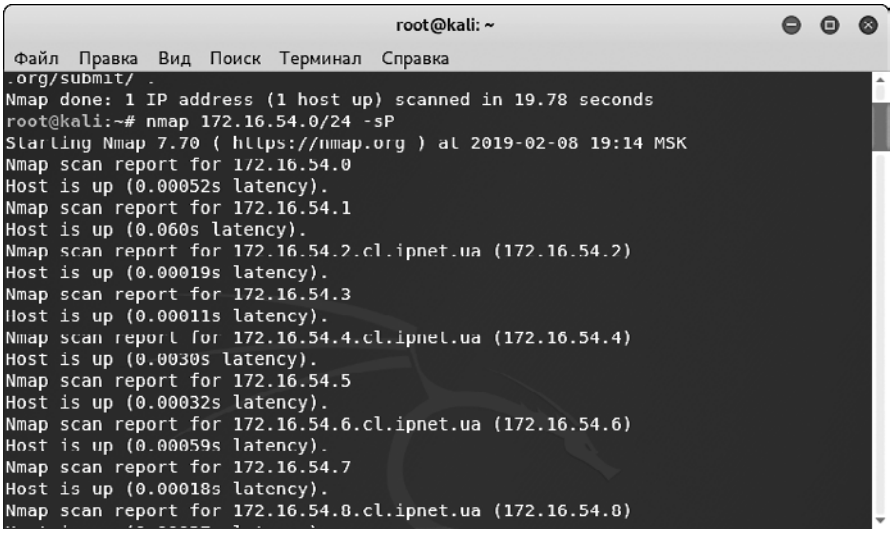

Рис. 3.3. Сканирование узла

Перечисление

Метод перечисления — это плацдарм для всех атак на слабые места, которые обнаруживаются в веб-приложениях. Все атаки на слабые места можно классифицировать по уязвимостям, которые появляются на разных этапах развития. Это может быть этап разработки, реализации или развертывания. Существует несколько методов перечисления. С некоторыми из них мы и познакомимся.

Совместное использование SBM

Server Message Block (SMB) обозначает блок сообщений сервера. Этот протокол обмена файлами был изобретен IBM в середине 1980-х годов и существует до сих пор. Назначение данного протокола — дать возможность компьютерам читать и записывать файлы на удаленный хост по *локальной сети (LAN)*. Каталоги на удаленных узлах SMB называются *акциями*.

Этот метод передачи данных имеет несколько преимуществ, которые мы и обсудим.

Передача зоны DNS. Протокол DNS — мой любимый протокол, потому что это просто кладезь информации. Данный протокол определяет связь имени хоста с IP-адресами всех хостов в сети. Если злоумышленнику известна схема сети, с помощью этого протокола он может быстро обнаружить все узлы в сети. С помощью DNS также можно создавать службы, работающие в сети, например почтовые серверы.

DNSRecon. Содержит инструменты разведки и перечисления. В этом примере мы запросим перенос зоны из домена domain.foo. DNS-сервер, работающий в домене domain.foo, вернет все записи, относящиеся к этому домену и ко всем связанным с ним поддоменам. Благодаря этой операции мы получим имена серверов, соответствующие им имена хостов и IP-адреса для домена. Будут возвращены все имеющиеся записи DNS: TXT-записи (4), PTR-записи (1), MX-записи для почтового сервера (10), записи протоколов IPv6 (2) и IPv4 (12). Эти записи действительно предоставляют пикантную информацию о сети. Одна запись показывает IP-адрес офиса DC, во второй записи вы увидите IP-адрес брандмауэра, в третьей — VPN и IP-адрес, и еще одна запись показывает IP-адрес почтового сервера и логин портала (рис. 3.4).

```
dnsrecon -d zonetranfer.zone -a
```

Здесь `-d` — домен; `-a` — выполнить перенос зоны.

SNMP-устройства

Простой протокол сетевого управления (Simple Network Management Protocol), сокращенно **SNMP**, используется для регистрации сетевых устройств и приложений и управления ими. SNMP можно применять для удаленной настройки устройств и приложений, но, если оставить его незащищенным, он также мо-

жет быть использован для извлечения информации об указанных приложениях и устройствах. Эта информация пригодится для лучшего понимания сети:

```
snmpwalk 192.16.1.1 -c PUBLIC
```

-c — это строка аутентификации устройства.

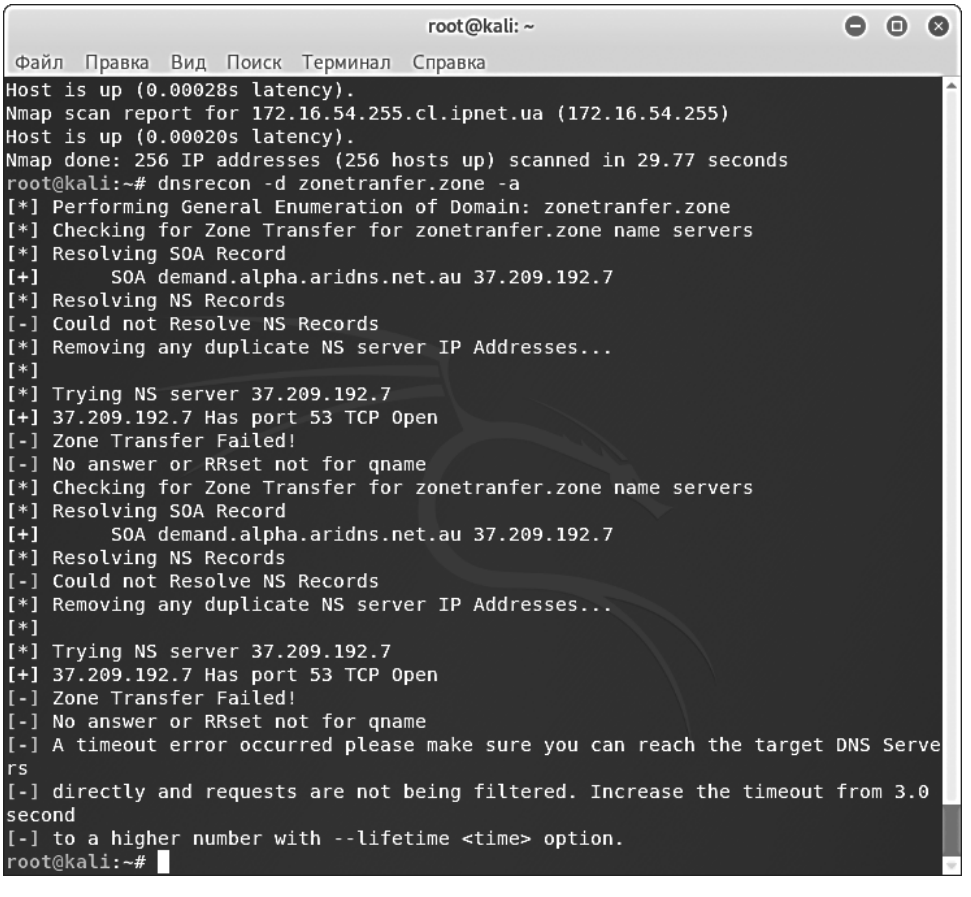

Рис. 3.4. Передача зоны DNS с помощью команды dnsrecon -d zonetranfer.zone -a

Захват пакетов

Захват пакетов, передаваемых между двумя хостами, может быть очень полезен при диагностике сетевых проблем, проверке учетных данных или, если вам нравится смотреть на пробегающий трафик, для развлечения.

tcpdump. Это утилита, которая запускается из командной строки и предназначена для прослушивания определенных типов трафика и передаваемых данных. Рассмотрим ее параметры:

- `-i eth0` — выбор интерфейса для прослушивания;
- `port 80` — выбор порта для прослушивания;
- `host 172.16.1.1` — только сбор трафика, идущего от хоста/к нему;
- `src` — данные приходят от хоста;
- `dst` — данные идут к хосту;
- `-w output.pcap` — захват трафика и сохранение его в файле на диске.

Wireshark. Утилита с графическим интерфейсом, используемая для прослушивания трафика на проводе (рис. 3.5):

- `ip.addr/ip.dst/ip.src == 172.16.1.1`;
- `tcp.port/tcp.dstport/tcp.srcport == 80`;
- `udp.port/udp.dstport/udp.srcport == 53`.

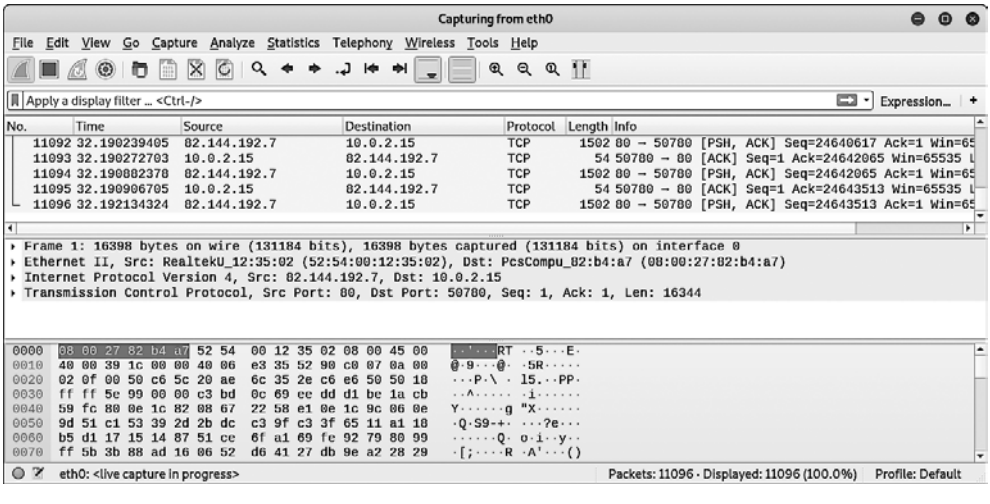

Рис. 3.5. Графическая утилита Wireshark

Получение доступа

Именно на этом этапе испытатели на проникновение пытаются закрепиться во внутренней сети компании. В настоящее время целевой фишинг (направленная атака на ваши персональные данные) — очень распространенный и эффективный способ достижения цели. Против организации может быть начата хорошо продуманная кампания по целевому фишингу с хорошо разработанным сценарием, основанным на информации, собранной ранее, на этапе разведки.

Получение доступа может также включать использование эксплойтов/учетных данных в удаленной службе для входа в систему и последующего выполнения полезных для исследователя нагрузок.

В этом вам могут помочь инструменты Metasploit и PowerShell Empire, поскольку оба создают полезные нагрузки, также известные как этапы. После запуска полезной нагрузки на целевом объекте процесс выполняется в памяти. Применение такого стиля позволяет оставлять очень мало улик. Другой вариант — передача бинарного файла в удаленную систему и его выполнение из командной строки, что также может быть эффективным. Данный подход быстрее, и его успешное выполнение не зависит от загрузки через Интернет.

Эксплойт

Иногда тестировщик находит сервисы, которые можно будет использовать. Эксплойт может послужить средством первоначального доступа. Вам лишь необходимо убедиться, что это средство надежно на 100 %. Но следует учесть, что неоднократный запуск эксплойта может привести к сбою в работе системы. Эту опцию нужно использовать очень осторожно и только в том случае, если вы ее протестировали и знаете, что с ней делать.

Эксплойтом может быть SSH! По крайней мере я никогда не видел, чтобы за пределами telnet использовалась другая служба.

Эксплойт для Linux

Эксплойты Linux обычно нацелены не на саму операционную систему, а на работающие в ней службы. Ниже приведен список распространенных эксплойтов для Linux:

- CVE-2018-1111;
- Red Hat Linux DHCP Client Found Vulnerable to Command Injection Attacks;
- CVE-2017-7494.

Эксплойт для Windows

Эксплойты Windows обычно нацелены на прослушивание служб операционной системы. Вот список, предназначенный для службы SMB, которая работает на порте 445 Windows:

- Eternalblue — MS17-010;
- MS08-67;
- MS03-026.

Ниже приведены некоторые инструменты, часто используемые испытателями на проникновение.

PsExec — инструмент из набора Sysinternals. Он используется для удаленного управления и популярен среди испытателей на проникновение, системных администраторов и хакеров.

Бинарный файл PsExec обычно копируется в общую папку `$admin` на компьютере, а затем использует удаленное управление для создания службы на удаленном компьютере. Имейте в виду, что PsExec на удаленной машине требует прав администратора.

1. Скачайте Sysinternals.
2. Запустите командную строку PowerShell.
3. С помощью команды `cd <Sysinternals directory>` создайте каталог `Sysinternals`.
4. Введите `.\PSexec \\<IP-адрес удаленной машины> -u <пользователь> -p <пароль> <cmd>`.

На рис. 3.6 показан полученный ответ.

Рис. 3.6. Ответ на введенную команду

Impacket — коллекция уроков Python для работы с сетевыми протоколами. Первоначальная настройка выполняется следующим образом:

1. Откройте терминал.
2. Введите `cd /tmp`.
3. Введите `git clone https://github.com/CoreSecurity/impacket.git`.
4. Введите `pip install`.

Для включения PSexec, WMI и SMBexec в Impacket используйте следующие команды.

- **PSexec:**

  `psexec.py <имя_пользователя>:<пароль>@<ip-адрес> powershell`

  Ответ на команду показан на рис. 3.7.

- **WMI:**

  `wmiexec.py <имя_пользователя>:<пароль>@<ip-адрес> powershell`

```
root@KaliLinuxVM:~/impacket# psexec.py student@10.80.100.142 powershell
Impacket v0.9.17-dev - Copyright 2002-2018 Core Security Technologies

Password:
[*] Requesting shares on 10.80.100.142.....
[*] Found writable share ADMIN$
[*] Uploading file mPlEodBY.exe
[*] Opening SVCManager on 10.80.100.142.....
[*] Creating service pwhM on 10.80.100.142.....
[*] Starting service pwhM.....
[!] Press help for extra shell commands
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\WINDOWS\system32> Get-Date
Get-Date

Wednesday, May 2, 2018 12:03:37 AM


PS C:\WINDOWS\system32>
```

Рис. 3.7. Ответ на команду psexec.py

На рис. 3.8 показан ответ на введенную команду.

```
root@KaliLinuxVM:~# wmiexec.py student@10.80.100.141
Impacket v0.9.17-dev - Copyright 2002-2018 Core Security Technologies

Password:
[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>
```

Рис. 3.8. Ответ на команду wmiexec.py

- **SMBexec:**

  `smbexec.py <имя_пользователя>:<пароль>@<ip-адрес>`

  Ответ на команду показан на рис. 3.9.

```
root@KaliLinuxVM:~# smbexec.py student@10.80.100.141
Impacket v0.9.17-dev - Copyright 2002-2018 Core Security Technologies

Password:
[!] Launching semi-interactive shell - Careful what you execute
C:\WINDOWS\system32>
```

Рис. 3.9. Ответ на команду smbexec.py

- **PS-Remoting.** Чтобы запустить PS-Remoting на целевом компьютере, выполните следующие действия:
  1) откройте на целевом компьютере от имени администратора PowerShell;
  2) введите следующее: `powershell -NoProfile -ExecutionPolicy Bypass -Command "iex ((new-object net.webclient).DownloadString('https://raw.githubusercontent.co /ansible/ansible/devel/examples/scripts/ConfigureRemotingForAn sible.ps1'))"`;
  3) включите PSRemoting;
  4) введите `winrm set winrm/config/client/auth '@{Basic="true"}'`;
  5) введите `winrm set winrm/config/service/auth '@{Basic="true"}'`;
  6) введите `winrm set winrm/config/service '@{AllowUnencrypted="true"}'`.

  Чтобы включить на целевой машине PS-Remoting, выполните следующие действия:
  1) откройте PowerShell;
  2) введите `$options=New-PSSessionOption -SkipCACheck -SkipCNCheck`;
  3) введите `$cred = Get-Credential`. Вам будет предложено ввести учетные данные;
  4) введите `Enter-PSSession -ComputerName <имя_хоста> -UseSSL -SessionOption $options -Credential $cred`.

  На рис. 3.10 вы увидите подробный ответ на введенную команду.

```
PS C:\> $options=New-PSSessionOption -SkipCACheck -SkipCNCheck
PS C:\> $cred = Get-Credential

cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:
Credential
PS C:\> Enter-PSSession -ComputerName 172.16.17.145 -UseSSL -SessionOption $options -Credential $cred
[172.16.17.145]: PS C:\Users\Sherlock Holmes\Documents> ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::103f:a1fe:34cd:a900%6
   IPv4 Address. . . . . . . . . . . : 172.16.17.145
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 172.16.17.2

Ethernet adapter Bluetooth Network Connection:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Tunnel adapter Teredo Tunneling Pseudo-Interface:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : 2001:0:4137:9e76:28de:3d40:53ef:ee6e
   Link-local IPv6 Address . . . . . : fe80::28de:3d40:53ef:ee6e%7
   Default Gateway . . . . . . . . . : ::
[172.16.17.145]: PS C:\Users\Sherlock Holmes\Documents>
```

Рис. 3.10. Реакция на команду Enter-PSSession

Подобным образом мы также можем включить WMI на удаленном целевом компьютере WMI для доступа к удаленной цели. Для этого запустите от имени администратора PowerShell и выполните следующую команду:

```
netsh firewall set service RemoteAdmin enable
```

Чтобы использовать WMI для доступа к удаленному целевому объекту, введите следующую команду:

```
wmic /node:<target IP addr> /user:<username> process call create "cmd.exe /c
<command>"
```

На экране появится ответ на нее в виде следующих данных (рис. 3.11).

Рис. 3.11. Ответ на команду wmic/node

Повышение привилегий

После получения доступа к целевой машине у вас будет низкий уровень привилегий. Учитывая, что задача любого испытания на проникновение состоит в имитации реальной атаки, которая включает в себя поиск конфиденциальной информации, хранящейся на серверах с ограниченным доступом, испытателю нужно будет найти способы повысить свои привилегии.

Поддержание доступа

После установки опорной точки (то есть получения удаленного доступа) при выходе пользователя из системы или перезагрузке компьютера ее можно быстро удалить. *Точка опоры* — это место постоянного доступа и входа. Установить ее можно несколькими способами. Наилучшей стратегией поддержания постоянного доступа является одновременное использование нескольких методов. Например, добавьте запасной вход (dropbox) в сеть, к которому позже можно будет получить доступ при наличии беспроводного подключения. Более хитрый способ поддержания доступа состоит в настройке запланированной задачи на взломанной машине, когда запуск происходит при перезагрузке, после чего задача периодически выполняется, например один раз в день (рис. 3.12).

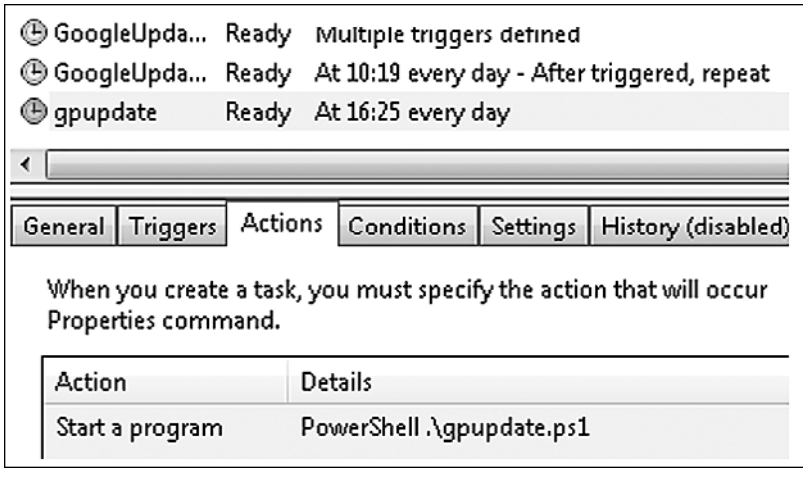

Рис. 3.12. Возможные точки доступа

Заметание следов

Еще раз отметим, что все ваши действия, несмотря ни на что, должны быть санкционированы клиентом. Но это не означает, что по окончании проверочного цикла, включающего в себя сканирование и эксплуатацию, испытатель идет домой. Необходимо составить отчет и представить результаты в понятной для заказчика форме. Но прежде чем приступить к составлению отчета, нам нужно очистить эксплойты и удалить инструменты, оставленные в рабочей среде. Это может означать удаление исполняемых файлов и редактирование журналов. Я говорю «редактирование», потому что любой системный администратор обязательно должен просматривать журналы. Иначе он может пропустить атаку. Поскольку операционные системы Windows и Linux имеют встроенные мощные средства ведения и документирования журналов событий, происходящих в операционной системе, о них мы рассказывать не будем. Я предлагаю вам отслеживать вносимые изменения и творчески редактировать журналы, когда вам нужно что-то скрыть. Используйте имена системных служб или имена пользователей, которые подходят для учетных записей. Например, не присваивайте учетной записи имя EliteHAK3R.

Составление отчета

Теперь мы подходим к финальной и, возможно, самой скучной части нашего теста. Однако, если вы следовали предыдущим этапам, отчетность не должна быть сложной и утомительной. Я пытаюсь делать заметки для отчета по мере прохождения теста и записываю промежуточные результаты или на бумаге, или с помощью встроенного в Kali инструмента Dradis, который вызывается командой `dradis start`. Имейте в виду, что это веб-сервис, поэтому любой человек на Земле,

зайдя по адресу `https://IP of kali machine:3004`, сможет получить к нему доступ. При первом же запуске Dradis вам будет предложено установить пароль.

Dradis позволяет импортировать файлы из Nmap, NESSUS, NEXPOSE и некоторых других текстовых редакторов. Это дает возможность делать заметки не только вам, но и вашим коллегам при командной работе. С помощью Dradis вы можете легко обмениваться информацией с товарищами по команде и фиксировать самые свежие результаты сканирования.

Резюме

Эта глава познакомила вас с различными методами испытания на проникновение. Полученные знания вы можете использовать, чтобы спланировать тест и определить области для проверки на проникновение. В следующей главе мы рассмотрим, как, используя не только пассивные, но и активные методы, обнаружить и собрать информацию о целевой среде и самой цели.